ĐẠI HỌC
BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY
OF SCIENCE AND TECHNOLOGY

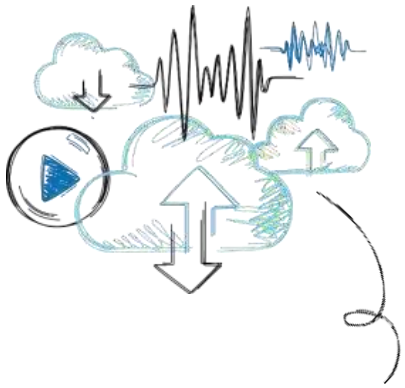
CS4451 – Computer Security

Basic Cryptography

Dr. Luu Quang Trung
trung.luuquang@hust.edu.vn

ONE LOVE. ONE FUTURE.

Objectives



1. Define cryptography
2. Describe hash, symmetric, and asymmetric cryptographic algorithms
3. Explain different cryptographic attacks
4. List the various ways in which cryptography is used

Defining Cryptography

- Defining cryptography involves:
  - Understanding what it is
  - Understanding what it can do
  - Understanding how cryptography can be used as a security tool to protect data

What is Cryptography? (1 of 7)

Cryptography (mật mã)

- Scrambling (*trộn*) information so it cannot be read
- Transforms information into secure form so unauthorized persons cannot access it

Steganography (giấu tin)

- Hides the existence of data
- An image, audio, or video file can contain hidden messages embedded in the file
- Achieved by dividing data and hiding in unused portions of the file
- May hide data in the file header fields that describe the file, between sections of the **metadata** (data used to describe the content or structure of the actual data)

What is Cryptography? (2 of 7)

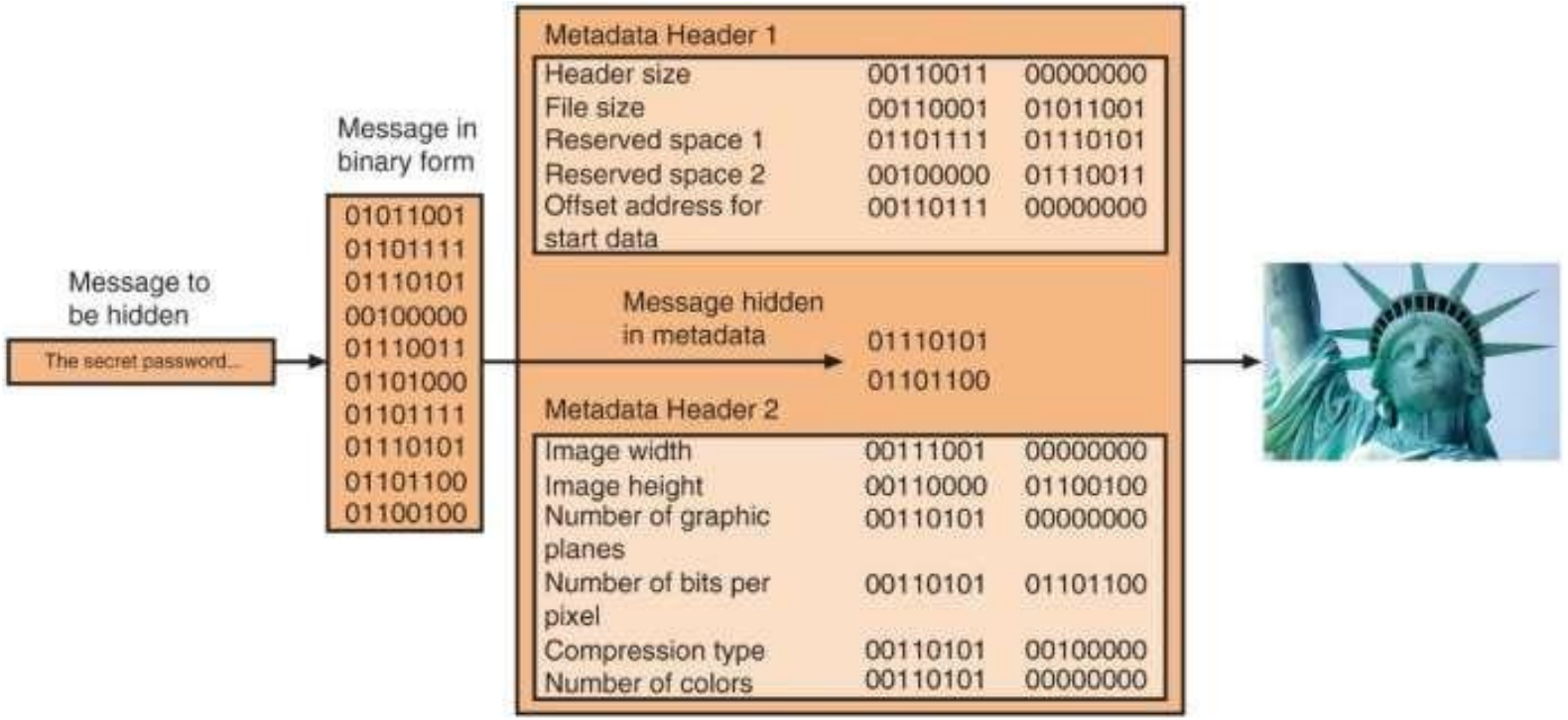


Figure 3-1 Data hidden by steganography

Photo: Chris Parypa Photography/Shutterstock.com

What is Cryptography? (3 of 7)

- Encryption
  - Changing original text into a secret message using cryptography
- Decryption
  - Changing secret message back to original form
- Plaintext
  - Unencrypted data to be encrypted or is the output of decryption
- Ciphertext
  - The scrambled and unreadable output of encryption
- Cleartext data
  - Data stored or transmitted without encryption

What is Cryptography? (4 of 7)

- Plaintext data is input into a **cryptographic algorithm (also called a *cipher*)**
  - Consists of procedures based on a mathematical formula used to encrypt and decrypt the data
- Key
  - A mathematical value entered into the algorithm to produce cipher text
  - The reverse process uses the key to decrypt the message
- Substitution cipher
  - Substitutes one character for another
  - One type is a ROT13, in which the entire alphabet is rotated 13 steps (A = N)
- XOR cipher
  - Based on the binary operation eXclusive OR that compares two bits

E = encryption = enciphering =/ encoding
D = decryption = deciphering =/ decoding
—> fishing
Y = E(K,X) —> X = D(K,Y) _ consistency

Y = K o+ X

a	b	(a o+ b)
0	0	0
0	1.	1
1	0.	1
1	1.	0

K o+ Y = K o+ (K o+ X)
= (K o+ K) o+ X
= O o+ X
= X

What is Cryptography? (5 of 7)

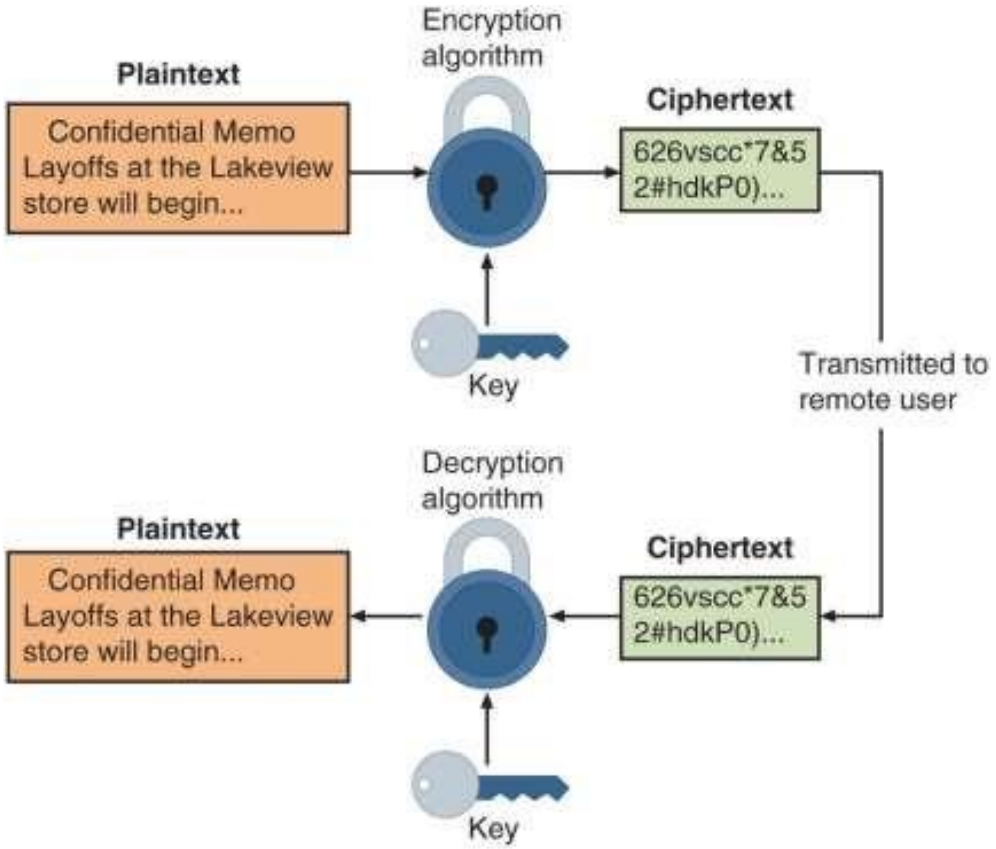


Figure 3-2 Cryptographic process

What is Cryptography? (6 of 7)

Substitution Cipher																										
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
Example	S	E	C	U	R	I	T	Y																		
Result	19	5	3	21	18	9	20	25																		
ROT13	A	B	C	D	E	F	G	H	I	J	K	L	M													
	N	O	P	Q	R	S	T	U	V	W	X	Y	Z													
Example	S	E	C	U	R	I	T	Y																		
Result	F	R	P	H	E	V	G	L																		

XOR Cipher	*a*	*b*		*a*	XOR	*b*
	0	0			0	
	0	1			1	
	1	0			1	
	1	1			0	

Example	S	E	C	U	R	I	T	Y
Combinator	F	L	A	P	J	A	C	K
Result	*150*	*90*	*20*	*51*	*80*	*81*	*71*	*2*

Figure 3-3 Cryptographic algorithms

Example [edit]

The string "Wiki" (`01010111 01101001 01101011 01101001` in 8-bit ASCII) can be encrypted with the repeating key `11110011` as follows:

```
  01010111 01101001 01101011 01101001
⊕ 11110011 11110011 11110011 11110011
= 10100100 10011010 10011000 10011010
```

And conversely, for decryption:

```
  10100100 10011010 10011000 10011010
⊕ 11110011 11110011 11110011 11110011
= 01010111 01101001 01101011 01101001
```

https://en.wikipedia.org/wiki/XOR_cipher

What is Cryptography? (7 of 7)

- Modern cryptographic algorithms rely upon underlying mathematical formulas
  - Depend upon the quality of random numbers (no identifiable pattern or sequence)
- Software relies upon a **pseudorandom number generator (PRNG)**
  - An algorithm for creating a sequence of numbers whose properties approximate those of a random number
- Two factors that can thwart threat actors from discovering the underlying key to cryptographic algorithms:
  - **Diffusion** – if a single character of plaintext is changed then it should result in multiple characters of the ciphertext changing
  - **Confusion** – the key does not relate in a simple way to the ciphertext

Cryptography and Security (1 of 3)

- Cryptography can provide five basic protections
  - **Confidentiality**
    - Ensures only authorized parties can view it
  - **Integrity**
    - Ensures information is correct and unaltered
  - **Authentication**
    - Ensures sender can be verified through cryptography
  - **Non-repudiation**
    - Proves that a user performed an action
  - **Obfuscation**
    - Making something obscure or unclear
- Security through obscurity
  - An approach in security where virtually any system can be made secure as long as outsiders are unaware of it or how it functions

Cryptography and Security (2 of 3)

Characteristic	Description	Protection
Confidentiality	Ensures that only authorized parties can view the information	Encrypted information can only be viewed by those who have been provided the key
Integrity	Ensures that the information is correct and no unauthorized person or malicious software has altered that data	Encrypted information cannot be changed except by authorized users who have the key
Authentication	Provides proof of the genuineness of the user	Proof that the sender was legitimate and not an imposter can be obtained
Non-repudiation	Proves that a user performed an action	Individuals are prevented from fraudulently denying that they were involved in a transaction
Obfuscation	Makes something obscure or unclear	By hiding the details the original cannot be determined

Cryptography and Security (3 of 3)

- Cryptography can provide protection to data as that data resides in any of three states:
  - Data in-use – data actions being performed by “endpoint devices”
  - Data in-transit – actions that transmit the data across a network
  - Data at-rest – data this is stored on electronic media

Cryptography Constraints (1 of 2)

- The number of small electronic devices (low-power devices) has grown significantly
  - These devices need to be protected from threat actors
- Applications that require extremely fast response times also face cryptography limitations
- **Resource vs. security constraint**
  - A limitation in providing strong cryptography due to the tug-of-war between available resources (time and energy) and the security provided by cryptography
- It is important that there be **high resiliency** in cryptography
  - The ability to quickly recover from these resource vs. security constraints

Cryptography Constraints (2 of 2)

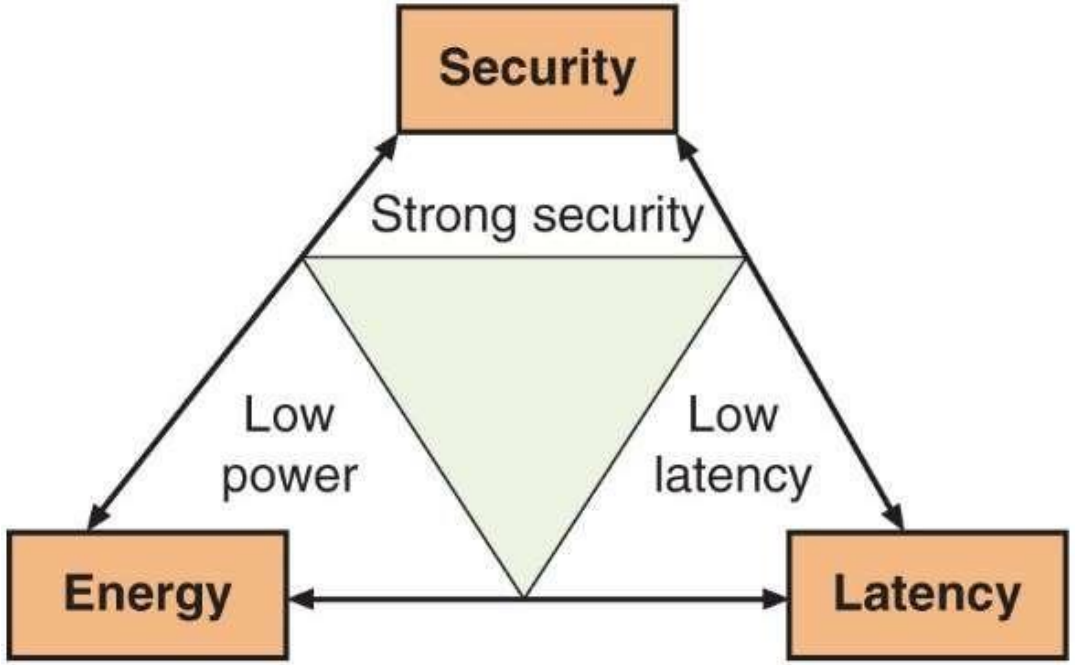


Figure 3-4 Resource vs. security constraint

Cryptographic Algorithms

- A fundamental difference in cryptographic algorithms is the amount of data processed at a time
  - **Stream cipher** - takes one character and replaces it with another
  - **Block cipher** - manipulates an entire block of plaintext at one time
  - **Sponge function** - takes as input a string of any length and returns a string of any requested variable length
- Three categories of cryptographic algorithms
  - **Hash** algorithms
  - **Symmetric cryptographic** algorithms
  - **Asymmetric cryptographic** algorithms

Hash Algorithms (1 of 5)

- Hash algorithms
  - Creates a unique *"digital fingerprint"* of a set of data and is commonly called **hashing**
  - This fingerprint, called a *digest* (sometimes called a *message digest* or *hash*), represents the contents
  - Its contents cannot be used to reveal original data set
  - Is primarily used for comparison purposes
- Hashing is intended to be one way in that *its digest cannot be reversed to reveal the original set of data*

Hash Algorithms (1 of 5)

Example:

- Input: "Xin chào"
- Hash algorithm: SHA-256
- Hash output:

"e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855"

Hash Algorithms (2 of 5)

- Secure hashing algorithm characteristics:
  - **Fixed size**
    - Short and long data sets have the same size hash
  - **Unique**
    - Two different data sets cannot produce the same hash
  - **Original**
    - Data set cannot be created to have a predefined hash
  - **Secure**
    - Resulting hash cannot be reversed to determine original plaintext

Hash Algorithms (3 of 5)

- Hashing is often used as a check to verify that the original contents of an item has not been changed

Image Name	Direct	Torrent	Size	Version	SHA1Sum
Kali Linux 64 bit	ISO	Torrent	2.9G	2016.2	25cc6d53a8bd8886fcb468eb4fbb4cdfac895c65
Kali Linux 32 bit	ISO	Torrent	2.9G	2016.2	9b4e167b0677bb0ca14099c379e0413262eefc8c
Kali Linux 64 bit Light	ISO	Torrent	1.1G	2016.2	f7bdc3a50f177226b3badc3d3eafcf1d59b9a5e6

Figure 3-5 Verifying file integrity with digests

Source: https://www.kali.org/downloads/

Hash Algorithms (4 of 5)

- **Message Digest 5 (MD5)**
  - Most well-known of the MD hash algorithms
  - Message length padded to 512 bits
  - Weaknesses in compression function could lead to collisions
  - Some security experts recommend using a more secure hash algorithm
- **Secure Hash Algorithm (SHA)**
  - More secure than MD
  - SHA-2 is currently considered to be a secure hash
  - SHA-3 was announced as a new standard in 2015 and may be suitable for low-power devices

Hash Algorithms (5 of 5)

- **Race Integrity Primitives Evaluation Message Digest (RIPEMD)**
  - The primary design feature is two different and independent parallel chains of computation
  - The results are combined at end of process
  - Several version of RIPEMD
    - RIPEMD-128, RIPEMD-256, and RIPEMD-320
- **Hashed Message Authentication Code (HMAC)**
  - A hash variation providing improved security
  - Uses a “shared secret key” possessed by sender and receiver
  - Receiver uses a key to decrypt the hash

Symmetric Cryptographic Algorithms (1 of 5)

- Symmetric cryptographic algorithms - use the same single key to encrypt and decrypt a document
  - Original cryptographic algorithms were symmetric
  - Also called **private key cryptography** (the key is kept private between sender and receiver)
- Common algorithms include:
  - Data Encryption Standard (DES)
  - Triple Data Encryption Standard (3DES)
  - Advanced Encryption Standard (AES)
  - Several other algorithms

Symmetric Cryptographic Algorithms (2 of 5)

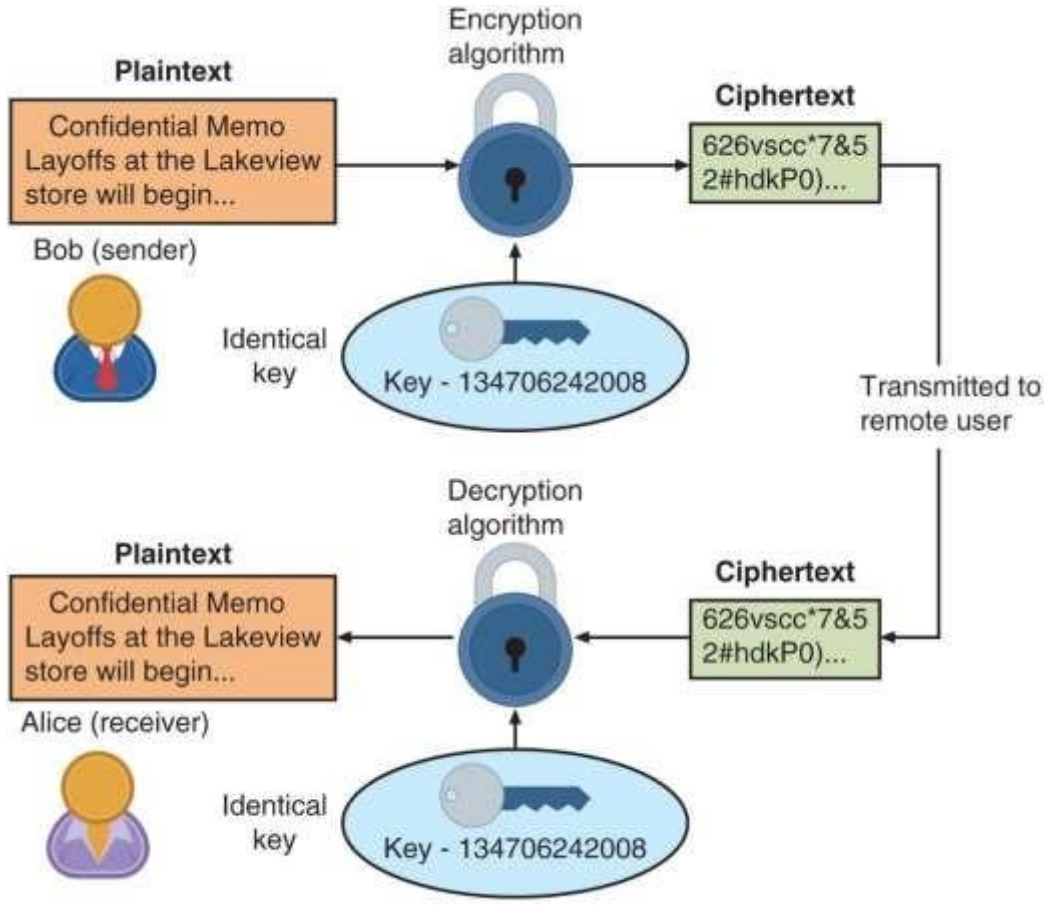


Figure 3-6 Symmetric (private key) cryptography

Symmetric Cryptographic Algorithms (3 of 5)

- **Data Encryption Standard (DES)**
  - Based on product originally designed in early 1970s
  - Uses a 56-bit key and is a block cipher
- **Triple Data Encryption Standard (3DES)**
  - Designed to replace DES
  - Uses three rounds of encryption
  - Ciphertext of first round becomes input for second iteration
  - Most secure versions use different keys used for each round

Symmetric Cryptographic Algorithms (4 of 5)

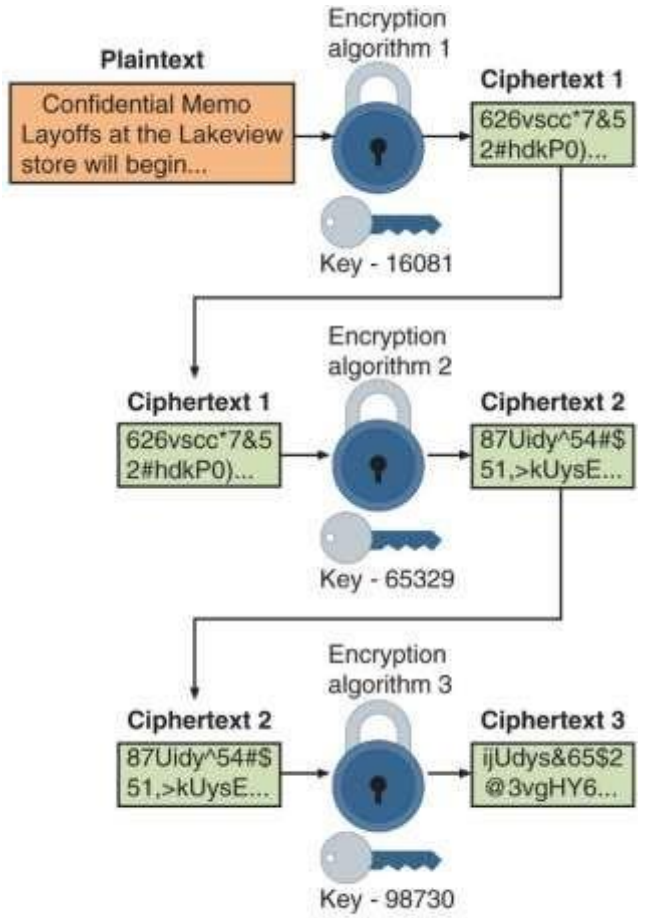


Figure 3-7 3DES

Symmetric Cryptographic Algorithms (5 of 5)

- **Advanced Encryption Standard (AES)**
  - A symmetric cipher approved by the NIST in 2000 as a replacement for DES
  - Performs three steps on every block (128 bits) of plaintext
  - Designed to be secure well into the future
- **Other Algorithms**
  - **Rivest Cipher (RC)**
    - Family of cipher algorithms designed by Ron Rivest
  - **Blowfish**
    - Block cipher operating on 64-bit blocks with key lengths from 32-448 bits
    - No significant weaknesses have been identified
  - **International Data Encryption Algorithm (IDEA)**
    - Used in European nations
    - Block cipher processing 64 bits with a 128-bit key with 8 rounds

Asymmetric Cryptographic Algorithms (1 of 8)

- Weakness of symmetric algorithms
  - Distributing and maintaining a secure single key among multiple users distributed geographically
- Asymmetric cryptographic algorithms
  - Also known as **public key cryptography**
  - Uses two mathematically related keys
  - Public key available to everyone and freely distributed
  - Private key known only to individual to whom it belongs

Asymmetric Cryptographic Algorithms (2 of 8)

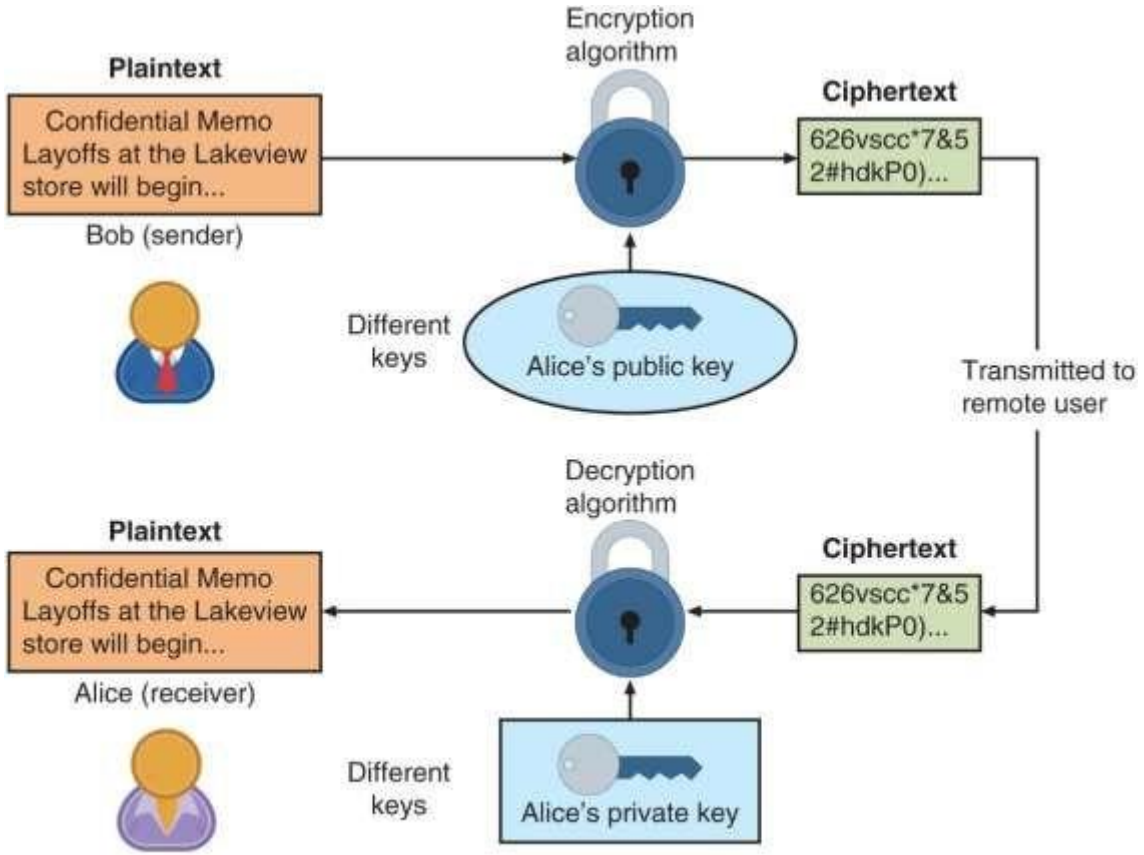


Figure 3-8 Asymmetric (public key) cryptography

Asymmetric Cryptographic Algorithms (3 of 8)

- Important principles
  - **Key pairs**
  - **Public key**
  - **Private key**
  - **Both directions** - keys can work in both directions
- Common asymmetric cryptographic algorithms:
  - RSA
  - Elliptic Curve Cryptography
  - Digital Signature Algorithm
  - Those relating to Key Exchange

Asymmetric Cryptographic Algorithms (4 of 8)

- **RSA**
  - Published in 1977 and patented by MIT in 1983
  - Most common asymmetric cryptography algorithm
  - Uses two large prime numbers
- **Elliptic curve cryptography (ECC)**
  - Users share one elliptic curve and one point on the curve
  - Uses less computing power than prime number-based asymmetric cryptography
    - Key sizes are smaller
  - Considered as an alternative for prime-number-based asymmetric cryptography for mobile and wireless devices

Asymmetric Cryptographic Algorithms (5 of 8)

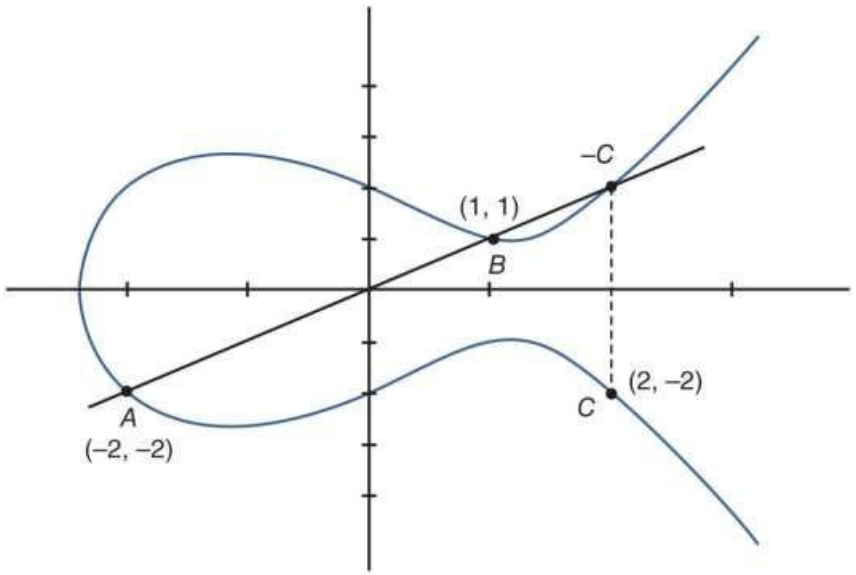


Figure 3-9 Elliptic curve cryptography (ECC)

1. Tạo khóa:

- **Chọn đường cong elliptic:** phù hợp với mức độ bảo mật
- **Chọn điểm cơ sở:** 1 điểm ngẫu nhiên trên đường cong
- **Tạo khóa bí mật:** 1 số nguyên ngẫu nhiên d
- **Tính toán khóa công khai:** Khóa công khai Q được tính toán bằng cách nhân điểm cơ sở với khóa bí mật $Q = d * G$

2. Mã hóa:

- **Chuyển đổi dữ liệu:** thành một điểm trên đường cong elliptic.
- **Mã hóa điểm:** sử dụng khóa công khai của người nhận:

$$C = M + k * Q$$

- **Gửi thông điệp:** Điểm mã hóa C và khóa bí mật tạm thời k được gửi cho người nhận.

3. Giải mã:

- **Tính toán điểm mã hóa:** Người nhận sử dụng khóa bí mật của mình để tính toán điểm mã hóa

$$M' = C - k * G$$

- **Lấy lại dữ liệu:** Dữ liệu ban đầu được lấy lại từ điểm mã hóa

$$M = M'$$

Asymmetric Cryptographic Algorithms (6 of 8)

- **Digital Signature Algorithm (DSA)**
  - Digital signature - an electronic verification
  - Verifies the sender
  - Prevents sender from disowning the message
  - Proves message integrity

Asymmetric Cryptographic Algorithms (7 of 8)

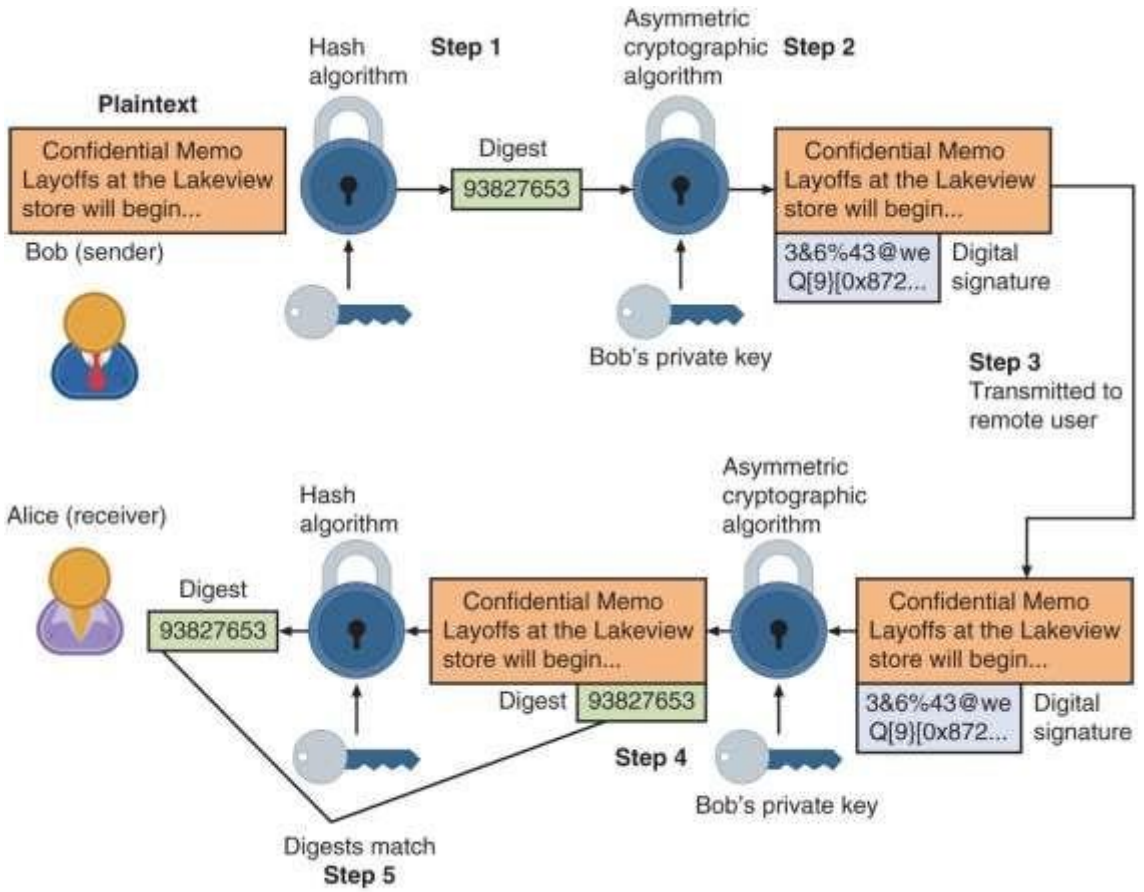


Figure 3-10 Digital signature

Asymmetric Cryptographic Algorithms (8 of 8)

- Key Exchange
  - There are different solutions for a key exchange that occurs within the normal communications channel (in-band) of cryptography:
    - **Diffie-Hellman (DH)**
    - **Diffie-Hellman Ephemeral (DHE)**
    - **Elliptic Curve Diffie-Hellman (ECDH)**
    - **Perfect forward secrecy**

Cryptographic Attacks

- Several of the more common cryptographic attacks include those that:
  - Target algorithm weaknesses
  - Exploit collisions

Algorithm Attacks (1 of 3)

- Methods attackers can focus on circumventing strong algorithms:
  - Known ciphertext attacks
  - Downgrade attacks
  - Using deprecated algorithms
  - Taking advantage of improperly implemented algorithms

Algorithm Attacks (2 of 3)

- **Known Ciphertext Attack**
  - Statistical tools can be used to attempt to discover a pattern in the ciphertexts, which can then be used to reveal the plaintext or key

Statistic	Example	How Used
Underlying language of plaintext	English	By knowing which language is used for the plaintext message inferences can be made regarding statistical values of that language
Distribution of characters	In English E is most commonly used letter, Q is least commonly used	Patterns can emerge when more common letters are used more frequently
Null ciphertexts	Distinguishing between actual ciphertexts and injected null messages	Attacks may inject a frame that contains null values to compare it with the frames containing ciphertext
Management frames	Analyze content of network management information	Because network management frames typically contain information that remains constant this can help establish patterns

Algorithm Attacks (3 of 3)

- **Downgrade Attack**
  - A threat actor forces the system to abandon the current higher security mode of operation and instead “fall back” to implementing an older and less secure mode
- **Using Deprecated Algorithms**
  - Means to use a cryptographic algorithm that should not be used because of known vulnerabilities
- **Improper Implementation**
  - Known as misconfiguration implementation
  - Many cryptographic algorithms have several configuration options
  - Unless careful consideration is given to these options the cryptography may be improperly implemented

Collision Attacks

- When two files have the same hash this is known as a *collision*
- **Collision attack**
  - An attempt to find two input strings of a hash function that produce the same hash result
- **Birthday attack**
  - Based on the *birthday paradox*, which says that for there to be a 50 percent chance that someone in a given room shares your birthday, 253 people would need to be in the room

- Ý tưởng chính của birthday attacks là sử dụng nguyên lý này để tìm ra một cặp giá trị đầu vào (thường là thông điệp, khóa hoặc bất kỳ dữ liệu nào khác) mà sau khi áp dụng một hàm băm (hash function), kết quả băm của chúng trùng nhau.
- Trong một tập hợp của n phần tử, xác suất để hai phần tử bất kỳ trùng nhau (có sự va chạm) là khoảng O(sqrt(n)), nơi O là ký hiệu Big O notation. Điều này có nghĩa là để có một khả năng va chạm xác đáng chú ý, chỉ cần một lượng dữ liệu tương đối nhỏ.
- Birthday attacks thường được áp dụng trong việc tấn công các hàm băm (hash functions), như SHA-1 hoặc MD5, khi mục tiêu là tìm ra hai thông điệp khác nhau nhưng có cùng giá trị băm (collision). Một khi collision được tìm thấy, nó có thể dẫn đến những vấn đề bảo mật nghiêm trọng, như giả mạo dữ liệu hoặc các cuộc tấn công phức tạp hơn. Đây là lý do tại sao các hàm băm mạnh mẽ và chống va chạm (collision-resistant) như SHA-256 hoặc SHA-3 thường được ưu tiên sử dụng trong các ứng dụng mật mã hiện đại.

Using Cryptography

- Cryptography should be used to secure:
    - Data-in-transit, data-at-rest, and when possible data-in-use
- This includes:
    - Individual files
    - Databases
    - Removable media
    - Data on mobile devices
- Cryptography can be applied through:
    - Software
    - Hardware

Encryption Through Software (1 of 2)

- **File and File System Cryptography**
  - Encryption software can be used to encrypt or decrypt files one-by-one
- **Pretty Good Privacy (PGP)**
  - Widely used asymmetric cryptography system
  - Used for files and e-mails on Windows systems
  - GNU Privacy Guard (GNuPG)
    - Open-source product that runs on Windows, UNIX, and Linux operating systems
  - OpenPGP is another open-source alternative that is based on PGP

Encryption Through Software (2 of 2)

- **Operating System Encryption**
  - Microsoft Windows Encrypting File System (EFS)
    - Cryptography system for Windows
    - Uses NTFS file system
    - Tightly integrated with the file system
    - Encryption and decryption are transparent to the user
- **Full Disk Encryption (FDE)**
  - Protects all data on a hard drive
  - Example: **BitLocker** drive encryption software that is included in Microsoft Windows
  - BitLocker encrypts the entire system volume, including the Windows Registry
  - Prevents attackers from accessing data by booting from another OS or placing the hard drive in another computer

Hardware Encryption (1 of 4)

- Software encryption can be subject to attacks to exploit its vulnerabilities
- Cryptography can be embedded in hardware
  - Provides higher degree of security
  - Can be applied to USB devices and standard hard drives
- Hardware encryption options include:
  - Trusted platform module
  - Hardware security model

Hardware Encryption (2 of 4)

- **USB device encryption**
  - Encrypted hardware-based flash drives can be used
    - Will not connect a computer until correct password has been provided
    - All data copied to the drive is automatically encrypted
    - Tamper-resistant external cases
    - Administrators can remotely control and track activity on the devices
    - Stolen drives can be remotely disabled
- **Self-Encrypting Drives (SEDs)**
  - Self-encrypting hard disk drives protect all files stored on them
  - The drive and host device perform authentication process during initial power up
  - If authentication fails, the drive can be configured to deny access or even delete encryption keys so all data is permanently unreadable

Hardware Encryption (3 of 4)

- **Trusted Platform Module (TPM)**
  - A chip on a computer's motherboard that provides cryptographic services
  - Includes a true random number generator
  - Entirely done in hardware so it cannot be subject to software attack
  - Prevents computer from booting if files or data have been altered
  - Prompts for password if hard drive moved to a new computer

Hardware Encryption (4 of 4)

- **Hardware Security Module (HSM)**
  - A secure cryptographic processor
  - Includes an onboard key generator and key storage facility
  - Performs accelerated symmetric and asymmetric encryption
  - Can provide services to multiple devices over a LAN

Chapter Summary (1 of 2)

- Cryptography is the practice of transforming information into a secure form while being transmitted or stored
- The strength of a cryptographic algorithm depends upon several factors
- Cryptography can provide confidentiality, integrity, authentication, non-repudiation, and obfuscation
- Hashing creates a unique digital fingerprint that represents contents of original material
    - Used only for comparison
- Symmetric cryptography uses a single key to encrypt and decrypt a message
    - Stream ciphers and block ciphers

Chapter Summary (2 of 2)

- Asymmetric cryptography
  - Public key cryptography
  - Uses two keys: public key and private key
- Cryptography can be applied through hardware or software
- Hardware encryption cannot be exploited like software cryptography